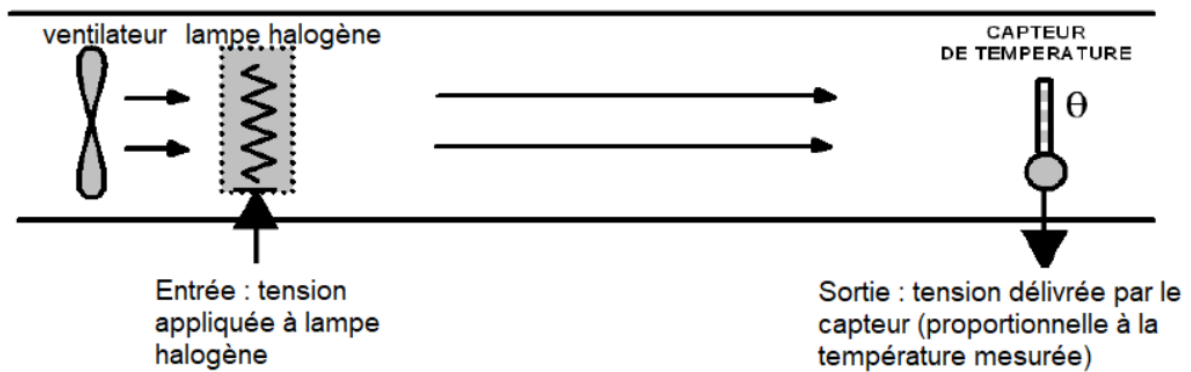


Lab 2 : Identification and control by state feedback of a thermal process

1 Description of the system and experimental data

The studied system is a thermal process model composed of:

- a halogen lamp: the system input is the voltage applied to this lamp (the heat released is proportional to this voltage).
- a temperature sensor: the output of the system is the potential difference provided by this sensor (proportional to the measured temperature)
- a fan whose intensity can be adjusted using a knob : it allows to modify the air flow between the lamp and the sensor, i.e. the way in which the input will influence on the output, or even the system parameters.



We interact with this system by using a PC equipped with an input/output card. To summarize, we remain the classical diagram which enables to study a continuous system via a computer and which is represented in Figure 1.

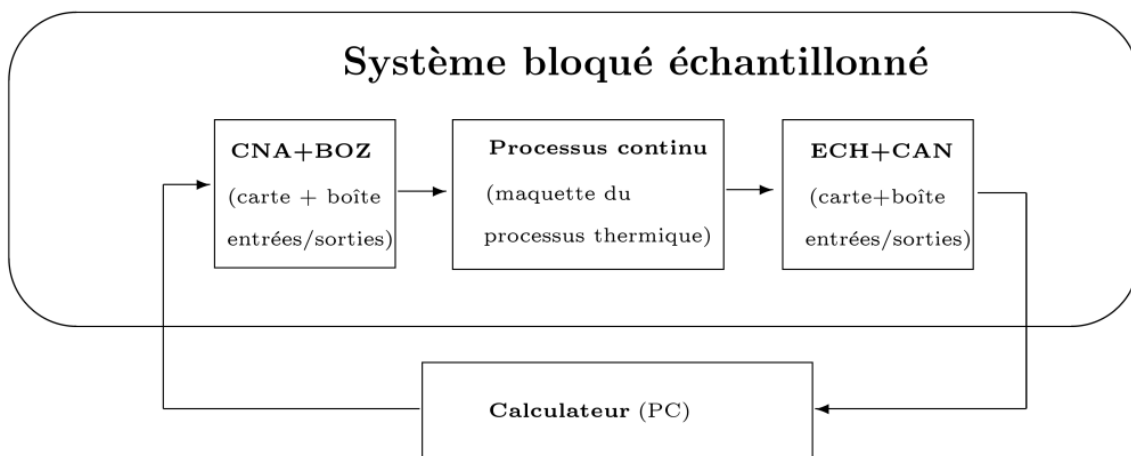


Figure 1 – Study diagram of the thermal process via a computer

Several experiments (excitation of the system by a step or a Pseudo-Random Binary Signal and measurement of its response) were carried out before the practical session and we will use the acquisition files that have been stored.

More precisely :

— The file called StepResponse.sod contains the data measured on the system when it is excited using a step.

— The files called PBRResponse1.sod and PBRResponse2.sod contain the data measured on the system during two experiments where the system was excited using a Pseudo-Binary Random Signal (PBR). Each of these files contains three vectors in the format expected by Scilab containing the sampling times (vector named t, in tenths of a second), the input samples (vector named u, in V) and the response samples in output (vector named y, in V). These data can be loaded into the Scilab environment via the command `load("FileName.sod")`.

2 System identification

2.1 Characterization

Qu. 1. Load the step response data into the Scilab work environment and plot the curve representing the response of the system y to the step as a function of time. Use the curve obtained to try to define the structure of your model: specify (with justification) the type of system (order, etc.).

Qu. 2 Use your Scilab script that implements the instrumental determinants ratio test using data from PBRResponse1.sod. Compare your conclusion to the previous question with the results obtained with this script.

Qu. 3 Propose a structure of the model of the sampled system in the form of a difference equation.

2.2 Parameter estimation

Qu. 4 Use your Scilab script which implements ordinary least squares to estimate the parameters of your difference equation using the data from PBRResponse1.sod.

Qu. 5 Use your Scilab script which implements recursive least squares using the data from PBRResponse1.sod to decide on the stationarity of the system during this experiment.

3 Feedback control

3.1 State representation

For a transfer function

$$H(z) = \frac{b_1 z + b_0}{z^2 + a_1 z + a_0},$$

one can use a state representation in the command canonical form given by:

$$\begin{cases} x(k+1) &= \begin{pmatrix} 0 & 1 \\ -a_0 & -a_1 \end{pmatrix} x(k) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(k) \\ y(k) &= \begin{pmatrix} b_0 & b_1 \end{pmatrix} x(k) \end{cases}$$

We can obtain this state representation by introducing a variable w as follows

$$H(z) = \frac{Y(z)}{U(z)} = \frac{Y(z) W(z)}{W(z) U(z)} = \frac{b_1 z + b_0}{z^2 + a_1 z + a_0},$$

With:

$$\frac{Y(z)}{W(z)} = b_1 z + b_0, \quad \frac{W(z)}{U(z)} = \frac{1}{z^2 + a_1 z + a_0}$$

then doing the inverse Z transform and setting

$$x(k) = \begin{pmatrix} w(k) \\ w(k+1) \end{pmatrix}.$$

Note that the variable $w(k)$ is “abstract” in the sense that its value does not directly reflect a physical quantity within the system. To simplify the following, we will nevertheless consider that the state variables $w(k)$ and $w(k+1)$ can be measured on the system. In the following, we use the Xcos application with Scilab (equivalent to Simulink with Matlab) to carry out the experiments. To try your hand, build the Xcos model shown in Figure 2 to simulate the step response of the system. Configure as follows:

- enter the matrices A, B, C and D as well as the initial value of the state vector in the state representation block;
- configure the Clock block: Period=1 and Initialization time=0
- configure the Scope block: Refresh Period=300, Ymin=-1, Ymax=3
- configure the simulation time (Simulation>Configure) to 300.

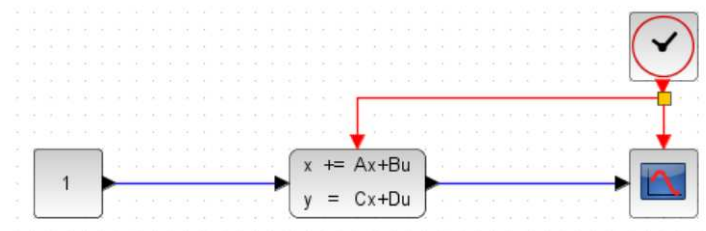


Figure 2 – Xcos model for simulating a step response of the model

Qu. 6 Run the simulation and check that you get a step response similar to the one plotted at the start of the lab. We will control the system using a command by status feedback. More specifically, we will apply the principle of pole placement, i.e. fix a regulation matrix L so as to impose the poles of the looped system, or even impose the characteristic polynomial of the corrected system

$|zI - A + BL|$ so that it is equal to

$$P_{des}(z) = z^2 + \alpha_1 z + \alpha_0.$$

Qu. 7 Check that the system is controllable.

Qu. 8 Express the regulation matrix L (as a function of a0, a1, α1 and α0).

Qu. 9 We also want to ensure a unity gain between the setpoint and the output. Determine the value of the gain Kre to be applied upstream of the closed loop as a function of (as a function of a0, a1, b0, b1, l1 and l0).

For a natural pulsation ωnd and a damping coefficient ξd (desired for our second-order system), the coefficients of the characteristic polynomial $P_{des}(z) = z^2 + \alpha_1 z + \alpha_0$ are given as follows :

$$\alpha_0 = e^{-2\xi_d \omega_{nd} \Delta},$$

$$\alpha_1 = -2e^{-\xi_d \omega_{nd} \Delta} \cos(\omega_{nd} \sqrt{1 - \xi_d^2} \Delta).$$

Here $\Delta = 1$.

Qu. 10 Calculate using a Scilab script the regulation matrix L and the gain Kre for ωnd = 0.5 and ξd= 0.9.

Qu. 11 Use Xcos to experiment with the status feedback command you have developed. To do this, build the Xcos model shown in Figure 3 to simulate the system equipped with status feedback:

- For the discrete-time state representation box: specify the matrices A, B for the evolution equation and an identity matrix as matrix C to be able to have the state vector output (pay attention to the dimensions of the zero matrix D).
- Specify C, L and Kre in the form of gain.
- Configure the clock with the following parameters: period = 1 and initialization time = 0.
- Configure the final integration time (Simulation>Configure) to 100. Simulate the corrected system. Compare the step response obtained with that obtained during the first simulation.

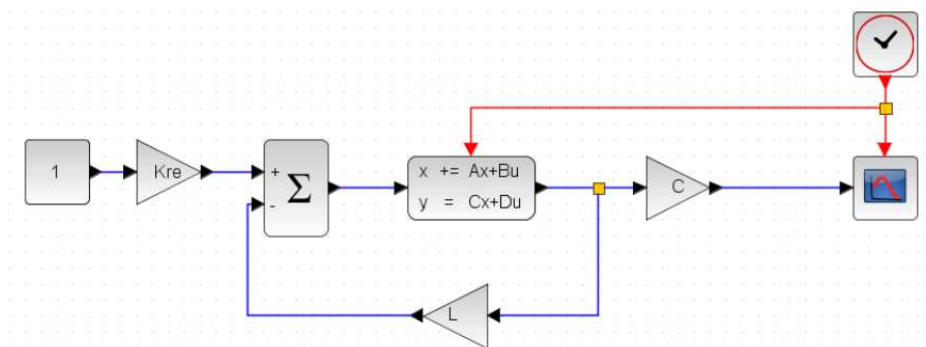


Figure 3 – Xcos model to simulate the system equipped with state feedback